What is claimed is:

1. An enterprise information system for managing data maintenance tasks across multiple applications within an enterprise, the system comprising:

a task manager module having an enterprise task list, said enterprise task list including tasks generated outside the system at the multiple applications and tasks generated within the system; and

a task engine module adapted to create tasks, said task engine module in communication with said task manager module.

2. An enterprise information system according to claim 1, wherein said task manager module further comprises predetermined task instructions for working each of the tasks.

3. An enterprise information system according to claim 1, wherein said enterprise task list includes a plurality of data fields and said task manager module is adapted to sort said enterprise task list according to any one or more of said data fields.

4. An enterprise information system according to claim 1, wherein tasks created within said task engine module are communicated to said task manager module for inclusion in said enterprise task list.

5. An enterprise information system according to claim 4, wherein tasks created by said task engine module are automatically worked by the system.

6. An enterprise information system according to claim 4, wherein tasks created by said task engine module are presented to be worked by a user.

7. An enterprise information system according to claim 1, wherein said task engine module automatically creates tasks according to defined rules.

8. An enterprise information system according to claim 7, wherein said task engine module mines data across the enterprise information system according to said defined rules to identify

missing or incorrect data and automatically creates tasks for correcting the missing or incorrect data.

9. An enterprise information system according to claim 2, wherein each of said predetermined task instructions includes one or more links to sites outside of the system.

10. A method of generating data maintenance tasks within an enterprise information system, comprising the steps of:

filing a source object and event to the system;

determining whether said event is a custom event or a system event;

adding all system events to an event queue;

determining for each of said custom events whether a corresponding custom event ruleset is true;

adding each of said custom events where said corresponding custom event ruleset is true to said event queue;

determining for each of said custom or system events in said event queue whether a task definition corresponding to each of said custom or system events exists;

determining for each task definition corresponding to each of said custom or system events whether a task definition ruleset is true; and

generating a task for each of said custom or system events having a task definition ruleset that is true.

11. A method according to claim 10, further comprising the step of adding said task to an enterprise task list.

12. A method according to claim 11, further comprising the step of determining whether said task is a duplicate task.

13. A method according to claim 10, wherein at least one of said tasks generated is worked by the system.

14. A method according to claim 10, wherein at least one of said tasks generated is presented to be worked by a user.

15. A method according to claim 10, wherein each of said tasks generated includes a predetermined set of instructions for working each of said tasks.

16. A graphical user interface for use in an enterprise information system for managing data maintenance tasks across multiple applications, the graphical user interface comprising:

a first portion including an enterprise task list, said enterprise task list including tasks generated outside the system at the multiple applications and tasks generated within the system.

17. A graphical user interface according to claim 16, further comprising:

a second portion including information regarding each of the tasks; and

a third portion including predetermined instructions for working each of the tasks.

18. A graphical user interface according to claim 17, wherein said predetermined instructions include one or more links to websites outside of the system.

19. A graphical user interface according to claim 17, wherein said predetermined instructions for working each of the tasks is sorted according to a predetermined workflow.

20. A graphical user interface according to claim 17, wherein said first, second, and third portions are included in first, second, and third frames, respectively.

21. A graphical user interface according to claim 17, wherein said second and third portions are included in layered, tabbed frames.

22. A system architecture for centrally managing the creation of tasks comprising:

a source object layer including one or more source objects, each of said one or more source objects having a corresponding event;

a target object layer including one or more target objects; and

an enterprise task manager layer between said source object layer and said target object layer, wherein said enterprise task manager layer centrally manages relationships between each of said one or more source objects and each of said one or more target objects.

23. A system architecture according to claim 22, wherein said enterprise task manager further comprises a ruleset manager module including an event queue.

24. A system architecture according to claim 22, wherein said ruleset manager receives each of said one or more source objects and corresponding event, determines whether one or more task definitions exist for each of said one or more source object and corresponding event, compares each of said one or more source object and corresponding event to predetermined task definition rulesets, and generates tasks to be performed on said one or more target objects based on said predetermined task definition rulesets.

25. A system architecture according to claim 22, wherein each of said corresponding events may be one of a system event and a custom event.

26. A system architecture according to claim 25, wherein each of said system events is automatically added to said event queue.

27. A system architecture according to claim 25, wherein each of said custom events is compared to a custom event ruleset to determine whether to add said custom event to said event queue.

28. A method of populating task data fields using task templates, the method comprising the steps of:

populating each of the task data fields that are empty with data from corresponding data fields in a task definition template;

populating each of the task data fields that are empty with data from corresponding data fields in a task name template;

populating each of the task data fields that are empty with data from corresponding data fields in a source object template; and

populating each of the task data fields that are empty with data from corresponding data fields in a system template.

29. A method according to claim 28, wherein the steps are performed in the following order:

(1) populating each of the task data fields that are empty with data from corresponding data fields in a task definition template;

(2) populating each of the task data fields that are empty with data from corresponding data fields in a task name template;

(3) populating each of the task data fields that are empty with data from corresponding data fields in a source object template; and

(4) populating each of the task data fields that are empty with data from corresponding data fields in a system template.

30. A method of automatically generating and performing tasks within an enterprise task management system, the method comprising the steps of:

providing an agent including a predetermined ruleset;

comparing data against said predetermined rules of said agent; and

causing the system to generate tasks according to said predetermined ruleset of said agent.

31. A method according to claim 30, wherein said comparing step is performed at predetermined time intervals.

32. A method according to claim 30, wherein a query including said comparing step may be performed at any time.

33. A method according to claim 30, wherein said data is one or more source object and event pairs filed to the system.

34. A method according to claim 30, wherein said data is task data.

35. A method according to claim 30, further comprising the step of:

providing an agent including a predetermined workflow for working a task; and

causing the system to automatically work the task according to said predetermined workflow of said agent.

36. An enterprise information system for managing data maintenance tasks across multiple applications, the system comprising:

an enterprise task list, said enterprise task list including tasks generated outside the system at the multiple applications and tasks generated within the system;

a means for the system to create and add tasks to said enterprise task list; and

a means for adding tasks created outside the system to said enterprise task list.

37. An enterprise information system according to claim 36, wherein said enterprise task list includes a plurality of data fields and a means for sorting said enterprise task list according to any one or more of said data fields.

38. A system architecture for centrally managing the creation of tasks comprising:

a source object layer including one or more source objects, each of said one or more source objects having a corresponding event;

a target object layer including one or more target objects; and

an enterprise task manager layer between said source object layer and said target object layer, said enterprise task manager layer having a means for centrally managing relationships between each of said one or more source objects and each of said one or more target objects.

39. A computer readable signal in which is contained a program for generating data maintenance tasks within an enterprise information system, the program comprising instructions that, when executed by a computer, perform the steps of:

filing a source object and event to the system;

determining whether said event is a custom event or a system event;

adding all system events to an event queue;

determining for each of said custom events whether a corresponding custom event ruleset is true;

adding each of said custom events where said corresponding custom event ruleset is true to said event queue;

determining for each of said custom or system events in said event queue whether a task definition corresponding to each of said custom or system events exists;

determining for each task definition corresponding to each of said custom or system events whether a task definition ruleset is true; and

generating a task for each of said custom or system events having a task definition ruleset that is true.

40. A computer readable signal according to claim 39, further comprising the step of adding said task to an enterprise task list.

41. A computer readable signal according to claim 40, further comprising the step of determining whether said task is a duplicate task.

42. A computer readable signal according to claim 40, wherein at least one of said tasks generated is worked by the system.

43. A computer readable signal according to claim 40, wherein at least one of said tasks generated is worked by a user.

44. A computer readable signal according to claim 40, wherein each of said tasks generated includes a predetermined set of instructions for working each of said tasks.

45. A computer readable signal in which is contained a program for populating task data fields using task templates, the program comprising instructions that, when executed by a computer, perform the steps of:

populating each of the task data fields that are empty with data from corresponding data fields in a task definition template;

populating each of the task data fields that are empty with data from corresponding data fields in a task name template;

populating each of the task data fields that are empty with data from corresponding data fields in a source object template; and

populating each of the task data fields that are empty with data from corresponding data fields in a system template.

46. A computer readable signal according to claim 45, wherein the steps are performed in the following order:

(1) populating each of the task data fields that are empty with data from corresponding data fields in a task definition template;

(2) populating each of the task data fields that are empty with data from corresponding data fields in a task name template;

(3) populating each of the task data fields that are empty with data from corresponding data fields in a source object template; and

(4) populating each of the task data fields that are empty with data from corresponding data fields in a system template.

47. A computer readable signal in which is contained a program for automatically generating and performing tasks within an enterprise task management system, the program comprising instructions that, when executed by a computer, perform the steps of:

providing an agent including a predetermined ruleset and a predetermined workflow for working a task;

comparing source object and event pairs filed to the system against said predetermined rules of said agent;

causing the system to generate tasks according to said predetermined ruleset of said agent; and

causing the system to automatically work the tasks according to said predetermined workflow of said agent.

48. A computer readable signal according to claim 47, wherein said comparing step is performed at predetermined time intervals.

49. A computer readable signal according to claim 47, wherein said comparing step may be performed at any time.